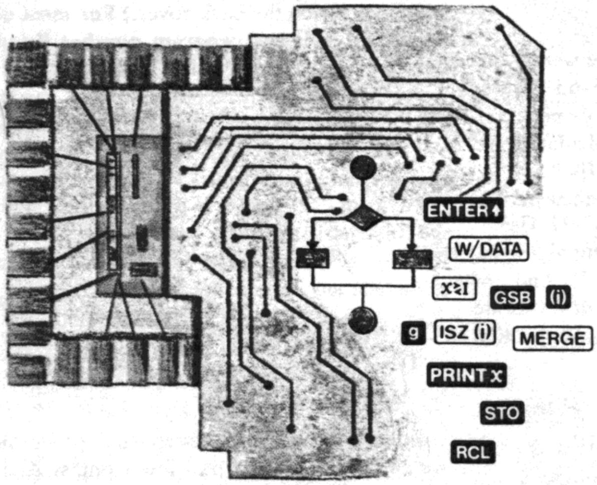


Featuring, this issue. . .

HEWLETT hp PACKARD

HP Key Notes

January 1977 Vol. 1 No. 1

Welcome Aboard!

For slightly over two years, we have been mailing a newsletter, HP-65 KEY NOTE , to HP-65 Users in the United States and in Canada. With the introduction of the HP-67 and HP-97, we decided to go into *worldwide* circulation of the newsletter. Hence: HP KEY NOTES.

As you look through the newsletter you will notice one concession to worldwide circulation: programs highlighted in HP KEY NOTES have two different numbers. This is done merely to satisfy certain computer peculiarities for the Library in the U.S. and the Library in Europe. It is very important that you *use the correct number and order from the right Library*. European Library services are extended to Users in Europe, North Africa, and the Middle East. All other countries must order programs from the Library in Corvallis, Oregon. The two addresses are listed on the back cover of HP KEY NOTES.

Notice that we now show the Library address as in Oregon. Since July of 1976 we have been gradually moving people and equipment to our new factory in Corvallis, Oregon. It has been a mammoth undertaking, and we hope that inevitable problems and delays caused by the move have not seriously inconvenienced anyone.

The photo shows our first completed building (on the right) and a second building nearing completion. (It will be finished in February 1977.) The buildings are on a 139-acre site northeast of Corvallis, which is in the beautiful Willamette Valley, about 80 miles south of Portland.

Although there are people and equipment that still must be moved to Oregon, for all practical purposes, we are here to stay, and (except for European areas) you should use only this address for all business.

Accessories Update

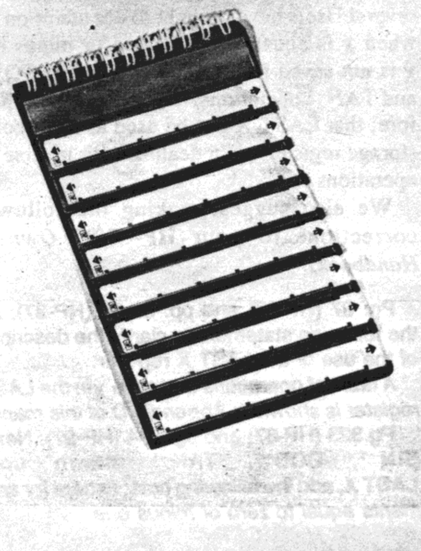
Because some of the accessories for the HP-67/97 calculators can be used for the HP-65, here is an up-to-date report on those accessories.

Foremost on HP-65 owners' "want list" has always been a better method for arranging, storing, and retrieving magnetic cards. Therefore you'll be happy to know that a new HP-67/97 accessory solves that problem. Pictured below is the new Program Card Holder that stores 40 magnetic cards in see-through plastic slots. There's also an open pocket behind each "page" of cards so that you can store notes, instructions, etc. The Program Card Holder can be ordered in three ways:

- A package of three Program Card Holders, model #00097-13142, for $10.
- One pack of 40 Blank Magnetic Cards plus a Program Card Holder model #00097-13141, for $20.
- Three packs of 40 Blank Magnetic Cards plus three Program Card Holders, model #00097-13143, for $45.

Another HP-67/97 accessory you will need in the future is the model #00097-13154 Programming Pad, for $4.00. It replaces all previous Programming Pads, including the one for the HP-65.

All of these—and other—accessories are available in the U.S. from your local HP dealer, or they can be ordered directly from the factory. *In all other countries,* these accessories are available from your HP dealer or from the nearest HP sales office. If you order by mail in the U.S., make checks or money orders payable to Hewlett-Packard and be sure to include any state or local taxes.



All prices in this newsletter are suggested retail prices excluding applicable state and local taxes—Continental U.S.A., Alaska and Hawaii.

User Feedback

Here are two items of interest to many of our HP-67/97 Users. They were brought to our attention, so we are passing them on to you to keep you well-informed and up to date.

SIN^{-1} AND COS^{-1} OF SMALL-MAGNITUDE AGRUMENTS

We want to let you know that several HP-67/97 Users have pointed out a few specific argument values for which sin^{-1} (and to a lesser degree, cos^{-1}) are in error to an extent that could be excessive for some applications. However, these arguments are very small in magnitude and thus infrequently encountered by most Users.

The six specific arguments affected and the resulting errors for sin^{-1} χ are: χ = 0.000003000 (error 0.6%), 0.000004000 (2.5%), 0.000005000 (4.0%), 0.000006000 (7.0%) 0.000007000 (8.0%), 0.000008000 (11.5%). No other values are affected. Notice that changing the magnitude of the above arguments by as little as ±0.000000001 eliminates the larger-than-normal error.

We suggest inserting a note regarding this on page 92 of your *HP-67 Owner's Handbook* or on page 83 of your *HP-97 Owner's Handbook.*

USING LAST X WITH INVERSE TRIG OPERATIONS

The LAST X register is primarily intended for error recovery in the event an undesired function is mistakenly executed; i.e., the "opposite" function can be easily executed on the same number, thus conveniently getting the User back to the previous step in the problem. Furthermore, with trigonometric functions, the *inverse* operation (SIN, SIN^{-1}) can be used to directly return to the previous step in the problem. Thus, LAST X is not essential for easy error recovery when dealing with trig functions.

In addition to this use of LAST X, some Users have found it useful to utilize LAST X as a temporary storage register. However, several Users have brought to our attention that when χ is equal to either zero or minus one, χ is not saved in LAST X for SIN^{-1}, COS^{-1}, and TAN^{-1} operations. We recommend, therefore, that LAST X **not** be used as a temporary storage register when dealing with inverse trig operations.

We also suggest making the following corrections to your HP-67/97 *Owner's Handbooks.*

Pg. 67 (HP-67) and pp. 58-59 (HP-97): Add the following statement to clarify the description of the use of the LAST X register.

A table of operations that save χ in the LAST X register is shown in Appendix D of this manual.

Pg.323 (HP-67) and pg. 294 (HP-97): Next to SIN^{-1}, COS^{-1}, TAN^{-1} shown under **LAST X**, add the following note: *Except for arguments equal to zero or minus one.*

Library Corner

As of December 15, 1976, there were 5,265 programs logged into the HP-65 Users' Library. As you can see, HP-65 owners continue to find new uses for their calculator, and the Library continues to steadily flourish.

As of December 16, 1976, there were 320 programs logged into the HP-67/97 Users' Library. You will find some of them highlighted under "NEW PROGRAMS." And, of course, a large number of initial programs are the individual programs from the Application Pacs.

EUROPEAN LIBRARY NEWS

Although very similar to the Library in the U.S., the European Library is actually part of two "clubs." The official names are "HP-67/97 Users' Club-Europe" and the same for the HP-65. The Library operates much the same as the one in the U.S., but differs in the charges for programs and so forth. Of course, there is a somewhat more critical problem there; a total of 14 different currencies are accepted by the Library. Maybe **that** is why the clubs are located in Switzerland!

There are three applications manuals (no magnetic cards) available in the European area. Two are in French; they are:

- *HP-67/97 Surveying Applications Manual* #00097- 90160
- *HP-67/97 Civil Engineering Applications Manual* #00097- 90161

The third one is in German; it is:

- *HP-67/96 Surveying Applications Manual* #00097- 90162

These three manuals, each of which contains 13 to 20 programs, are available through your nearest HP sales office or dealer in the German and French speaking areas of Europe.

German Civil Engineering Software Available

It should be of interest to all Civil Engineers in German-speaking areas that an extensive range of software is available through the two addresses noted below. Both of these sources have already contributed many programs for the HP-65 and have aquired a very good reputation among their colleagues.

Engineer Bureau Heiniger
Feldweg 8
Wetzikon/Zurich
Switzerland

Engineer Bureau Weckmann
Am Reulert 6
5100 Aachen
Germany

ORDERING PROGRAMS

Any program you see in HP KEY NOTES can be ordered from either the Users' Library in Corvallis, Oregon, or from the Users' Library in Geneva, Switzerland. (Both addresses are on the back cover.) **For most of the world, use the program number listed next to the program's title,** and order from Corvallis. The only exception is if you live in the European areas; in that case, *use the number listed in italic type below the program abstract,* and order from Geneva.

Payment for programs must conform with the instructions from *your* Library area. Always use order forms if possible, and be sure to include any state or local taxes.

NEW PROGRAMS

Following are some interesting programs we've received in the past few months. And, because the HP-67 and HP-97 are in the limelight at present, all the programs listed here are for those calculators. It is also significant that most of the first batch of programs submitted to the Library were games. So that is what you will find here.

We have not listed new HP-65 programs because a new Catalog Update of approximately 2,000 programs was just released. There are, however, two HP-65 programs featured on later pages of this issue. We will feature some new HP-65 programs—perhaps some from Europe or Asia—in the next issue of HP KEY NOTES.

Concentration (00266D)*

The player tries to memorize a random number that is displayed for only 2½ seconds. If you are successful, a larger number will be displayed; if not, a smaller number will be displayed. Credit is given for each correct digit, and the game ends after three misses. Your score is the total number of correct digits, and it is represented by a description of your ability to remember numbers. It can range from a low ability of "total amnesia" to a high of "eidetic memory" on the scoring chart. Also, a difficulty factor can be entered at the start of any game. (111 steps)

Author: **William M. Kolb**
Upper Marlboro, Maryland

**In European areas, order by number 00195D.*

Shooting Gallery (00219D)*

Step right up to the arcade, drop a quarter in the old shooting gallery machine, and try to hit the moving decimal point. You have 12 shots (2 six-guns), but the machine can shoot back! Try to beat the 200-point limit for a free game, but don't let the machine get six or more shots off *at you* or you lose! (140 steps)

Author: **Craig A. Pearce**
Berwyn, Illinois

**In European areas, order by number 00201D.*

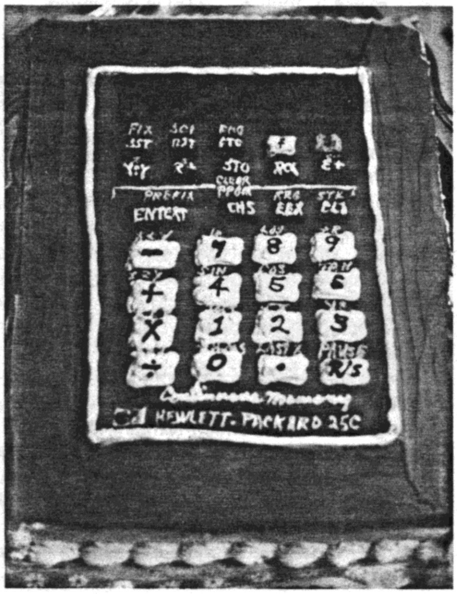
GTO *page* 12

A High-Calory Calculator

Although this is not a programming trick or tip, it *is* a "far-out" application of HP calculators. Besides, we figured it would both interest and amuse you, and maybe give your spouses an idea for *your* birthday.

Saturday night, October 30, turned out to be quite a surprising time for **Andy Burg** of Los Angeles, California. His fiancée, his sister, and some friends had a surprise birthday party for him, and the birthday cake was, to say the least, most extraordinary—as you can see. What you can't see, however, is the amount of consternation caused by the cake. It took three tries by the baker before it came out anywhere near correct! The final layout was accurate except for the "g" functions. But if Mr. Burg's sister had trouble getting an HP-25C cake correct, can you imagine the problem she would have had with the multitudinous functions on the HP-67?

Anyway, the cake was a huge success, and it did feed 30 people at the surprise costume party. And we thank Mr. Burg for sharing the photo with all of our readers.



Converting HP-65 Programs To The HP-67/97

If you have a lot of HP-65 programs and have purchased an HP-67 or HP-97, you probably want to know how to convert your old programs so you can use them with your new calculator. Remember: Although HP-65 and HP-67/97 blank magnetic cards are interchangeable, *you cannot run an HP-65 program in an HP-67/97, or vice versa*. The HP-65 uses a 6-bit word to represent program steps and the HP-67/97 uses an 8-bit word to do the same thing. (Also, all operations in the HP-67/97 are merged into single program steps.)

So, for the above reasons, there are some things to remember when you convert programs. And, although it is usually advantageous to redevelop and rewrite HP-65 programs to utilize the added features of the HP-67/97, many HP-65 programs can be directly converted to the HP-67/97 by using the following guidelines.

1. A subroutine on the HP-67/97 is called with the GSB key.

```
HP-65        HP-67/97
 LBL
  A           LBL A
  :             :
 f⁻¹
  ln            eˣ
  B           GSB B
  :             :
 RTN           RTN
```

   This also illustrates that all commands on the HP-67/97 are merged.

2. Keys f^{-1} SF1 and f TF1 on theHP-65 have been replaced by CLF 1 and F? 1, respectively, on the HP-67/97. Flag 2 on the HP-65 should be replaced by flag 0 on the HP-67/97. This is because flag 2 on the HP-67/97 clears each time it is tested.
3. There is no HP-67/97 test for flag off (f^{-1} TF1 on the HP-65). You should test for flag on, and then reverse the order of the next two commands. For example:

```
HP-65                   HP-67/97
 :                        :
f⁻¹   Is flag 1         F?1    Is flag 1 on?
TF1   Off?              GTO 0  Yes, then GTO 0.
GTO   Yes, then         GTO 1  No, then GTO 1.
1     GTO 1.            LBL 0
2     No, multiply      2
x     by 2 and          x
 :    continue           :
```

4. Conditionals skip only 1 step, instead of 2, when not satisified. Therefore, in some instances, it may be necessary to call a subroutine in an HP-67/97 program. For example:

```
HP-65              HP-67/97
:                  :
g x≤y              x≤y?
1 Yes, add 1 and   GSB 0 Yes, execute      LBL 0
                         routine 0         1
+ continue.                                +
2 No, skip two     2                       RTN
x steps.           x
:                  :
```

5. In the HP-65, pressing a user-definable key (A through E) when no such label exists in program memory, causes program execution to begin from the top of memory. Similarily, a GTO ⟨label⟩ within a program when the ⟨label⟩ does not exist transfers control to the top of program memory. On the HP-67/97, every label called must be specified, otherwise **Error** will appear on the display.
6. If an *undefined* subroutine (say, C) is called in a running HP-65 program, then C will be ignored and the first RTN encountered will act like a NOP. When an undefined subroutine is called on the HP-67/97, **Error** will appear on the display.
7. If a series of label-return programs are in program memory, and execution is started with R/S, the HP-65 will pass through the first return encountered and stop at the second. The HP-67/97, in the same situation, will stop at the first RTN.
8. The HP-67/97 does not halt on underflow; it places 0 in the display or register and continues program execution.
9. The HP-67/97 does not have a NOP (no operation) statement. If it is necessary to have a "filler" step, you may insert a label (e.g., LBL 9), which is never called.
10. The display formatting of a program is "remembered" on the HP-67/97 magnetic card. The HP-65 command for displaying four decimal places is DSP • 4. On the HP-67/97, this is accomplished by pressing FIX DSP 4.
11. There is no D.MS− key on the HP-67/97. Instead, change the sign (CHS) of the angle (or time) in the display and add with H.MS+.
12. There is no CLEAR STK key on the HP-67/97 because it is usually not essential to program operation. However, if the need arises, pressing CLx ENTER↑ ENTER↑ ENTER↑ gives the same results.
13. The HP-65 uses R_8 for the DSZ operation, while the HP-67/97 uses I, or can decrement and skip indirectly. The following program sums the integers from 1 to n in decreasing order. Simply key in the value of n and press A.

```
HP-65                    HP-67/97
LBL
A                        LBL A
STO 8  n → R8            STO I
0      0 → x             0
LBL                      LBL 1
1                        RCL I
RCL 8                    +
+      x + R8 → x        DSZ I
g                        GTO 1
DSZ    R8 − 1 → R8       RTN
GTO
1
RTN    Display sum
       (x)
```

14. There is no decimal ⇆ octal key on the HP-67/97. If you need this function, you should get program #00173D, *Hex-Octal Arithmetic*, from the HP-67/97 Users' Library.
15. The key f^{-1} on the HP-65 means "do the *opposite* of the next step." For instance, f^{-1} √x means x^2. f^{-1} INT means FRAC, and f^{-1} LOG is 10^x.

Equation Keystrokes ONLY!!

(The following program-editing technique was developed by **C. Ray Kolker,** *Engineering Manager for Faxon Communications Corp. in Pasadena, California. Although Mr. Kolker's contribution is copyrighted, he felt it should be shared with the readers of HP KEY NOTES, and we agreed. Therefore, he gave us permission to print his technique for our readers all over the world. Ed.)*

EQUATION KEYSTROKES *ONLY* REQUIRED TO PROGRAM YOUR HP-65/67/97 PROGRAMMABLES*

You bought a programmable calculator to save precious time and energy while solving math problems, those ranging from the simple to the very complex. Right? Now, would you like a way to save time and energy in the *use* of your programmable calculator?

I would like to introduce you to a simple, but extremely versatile, time-saving method of program writing. The expressions $E = mc^2$, $C^2 = A^2 + B^2$, $d = rt$, $X_L = 2\pi fL$, $1/f = 1/p + 1/q$, and $DP = N/PD$ are but a few of the common, often-used three-variable equations. All require exactly the same control-logic or program control instruction key codes. So, why not generate a simple, *universal,* control-logic program card; read it into program memory; key into memory only the equation keystrokes; then record the entire program (control-logic + equations) onto another program card? For instance:

Key this into memory: (HP-67/97)

```
001 f LBL A    008 f LBL B    015 f LBL C
    STO A          STO B          STO C
    h F? 3         h F? 3         h F? 3
    R/S            R/S            R/S
┌ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ─ ┐
└ ─ f LBL 1 ─ ─ ─ f LBL 2 ─ ─ ─ f LBL 3 ┘
    STO A          STO B  /       STO C
007 R/S        014 R/S   /    021 R/S
                        /
LOCATING-LABELS        /
FOR INSERTING
EQUATIONS
```

Record onto a program card:

You have just recorded the control-logic for interchangeable solutions for *all* three-variable equations in *any* discipline!! (Solves for: A in terms of B and C, B in terms of A and C, and C in terms of A and B.) Once you have recorded this card, it can be used over and over again.

The hard part of programming is designing the program control-logic keystroke sequence. So, now that that's behind us (forever, we hope, for this type problem; why keep reinventing the wheel?), let's key in the easy part: the equation keystrokes.

EQUATION ENTRY INSTRUCTIONS

PYTHAGOREAN THEOREM SAMPLE EQUATIONS AND KEYSTROKE SEQUENCES

$A = \sqrt{C^2-B^2}$ $B = \sqrt{C^2-A^2}$ $C = \sqrt{A^2+B^2}$

```
Equation "A"   Equation "B"   Equation "C'
Keystrokes     Keystrokes     Keystrokes
 RCL C          RCL C          RCL A
 g x²           g x²           g x²
 RCL B          RCL A          RCL B
 g x²           g x²           g x²
 −              −              +
 f√x            f√x            f√x
```

1. In RUN mode, insert program card with three-variable, interchangeable-solution, control-logic program.

2. Enter equations.
Equation "A"—In RUN mode, press GTO 1. Switch to W/PRGM mode. Code 31 25 01 is displayed. Key Equation "A" keystrokes into program memory. Switch to RUN mode.
Equation "B"—In RUN mode, press GTO 2. Switch to W/PRGM mode. Code 31 25 02 is displayed. Key Equation "B" keystrokes into program memory. Switch to RUN mode.
Equation "C"—In RUN mode, press GTO 3. Switch to W/PRGM mode. Code 31 25 03 is displayed. Key Equation "C" keystrokes into program memory.

3. Record onto a program card:

It's done!! The program is loaded, recorded, and ready to use!

TESTING THE FINISHED PROGRAM

In RUN mode, from the keyboard, perform the following operations:

a) Press 3, then A. 3.00 is displayed.
Press 4, then B. 4.00 is displayed.
Press C. Answer 5.00 is displayed.

Notice that input data is entered through two user-defined keys, such as A and B, and the answer is obtained by pressing C.

b) Press B. 4.00 is displayed.
Press A. 3.00 is displayed.

Notice here that this test verifies that each equation is properly entered into program memory. Actually, when you pressed B, then A, you solved for each of them based on the original solution you made for "C" in step a. (All variables, including the answer variable, were stored in step a.)

Since all variables are stored, it is then convenient to modify data without the necessity for keying in that data which remains the same. If you wish to change only the value of "B", ("A" remaining the same), merely key in a new value from the keyboard (digits keys) and press B. Press C for a new answer.

To see how interchangeable solutions work, solve for:
(Note: Let A = 3, B = 4, C = 5.)

1) C, entering A first then B, press C for answer (step a).
2) C. entering B first then A, press C for answer.
3) A, entering B first then C, press A for answer.
4) A, entering C first then B, press A for answer.
5) B, entering A first then C, press B for answer.
6) B, entering C first then A, press B for answer.

This illustrates that data may be entered in any order; i.e., "A" first, then "B", or "B" first, then "A", etc.

Observe the following:

a) Display information, such as DSP 9, h ENG, etc., may be entered in RUN mode before recording the completed program onto a program card.

**Copyright 1976 by C. Ray Kolker*
Faxon Communications Corp.
Pasadena, California 91106

(continued)

b) Locating-labels, LBL's 1, 2, 3, etc., may optionally remain, providing convenience in locating the beginning of each equation in the finished program by pressing GTO 1, GTO 2, GTO 3, etc. in RUN mode, or may be deleted if they are required for other uses.

c) If erroneous data is entered, the new data entered will write over the old data. Initialization is never required!!

d) Arithmetic operations may be performed just prior to pressing an "answer" key, but this will make the "answer" key a data input key when it is pressed. Pressing the "answer" key again will make it an "answer" key providing the correct answer.

e) Flag 3, and as many storage registers as there are variables are used.

THERE'S MUCH MORE!!

This program is truly flexible. Look at this!! (Note: Refer to equation entry instructions.)

1) Suppose, using the Pythagorean Theorem, you wish only to solve for "C". Merely insert equation "C" into LBL C, using GTO 3. The A and B keys are then data input keys only. C is the "answer" key.

2) If you need interchangeable solutions of a two-variable equation such as *Volume of a Sphere,* $V = 4\pi r^3/3$, solve algebraically for "r" in terms of "V", then insert equation "r" into LBL A using GTO 1 and equation "V" into LBL B using GTO 2. To use this, enter data through key A or B. B or A becomes an "answer" key. ("LBL C is not used.)

3) If you wish to solve one, two, or three independent, two-variable equations such as *Volume of a Sphere,* just insert equation "V" into LBL A using GTO 1. To use this, merely enter data via key A, then press A again for the answer. The other one- or two-equations would be inserted into LBL's B and C respectively.

4) Or suppose you need interchangeable solutions of a five-variable equation, or solutions of two independent equations, one being a three-variable and the other a two-variable equation (such as Pythagorean Theorem and *Volume of a Sphere*)? Simply expand the three-variable interchangeable-solution program to a five-variable program by adding LBL's D and E as follows:

a) In RUN mode, insert the program card with the three-variable, interchangeable-solution, control-logic program.

b) Press GTO.021. Switch to W/PRGM mode. Step 021, code 84 is displayed.

c) Key this into memory:

```
022 f LBL D     029 f LBL E
    STO D           STO E        LOCATING-
    h F? 3          h F? 3       LABELS
    R/S             R/S
    f LBL 4         f LBL 5
    STO D           STO E
028 R/S         035 R/S
```

LBL's A, B, C, D, and E can be used for a five-variable interchangeable-solution problem, or LBL's A, B, and C can be used for the Pythagorean Theorem, while LBL's D and E are used for *Volume of a Sphere.*

Interchangeable solutions of up to 10 variables are easily accommodated on the HP-67/97 by simple expansion of this basic program! A similar programming technique can also be accomplished on the HP-65.

Enjoy enlarging your personalized program library with this program writing approach: EQUATION KEYSTROKES *ONLY*!!

Users Club Grows

In just 2½ years, a dedicated group of calculator enthusiasts, inspired and led by **Richard J. Nelson,** have built the *HP-65 Users Club* into a close-knit organization that now includes Users of HP-25/25C/55/65/67/97 programmable calculators.

The club is a volunteer, non-profit, loosely organized, independent, worldwide group of people (now over 1500) who own and use Hewlett-Packard programmable calculators.

The primary objective of the club is to provide a means for HP programmable calculator Users to share their questions, problems, experiences, programs, and professional expertise for the common goal of obtaining the maximum use from their calculators. The club has three main activities:

- Monthly Newsletter. A 10-page (minimum) newsletter called *65 NOTES.*
- Membership Identification. Members receive a membership list, which facilitates locating people with similar interests.
- Program Sharing. Members may participate in individual program exchange by adding their program titles and statistics to the "Share-a-Program" data base.

Local chapters have been formed in many U.S. cities and in foreign countries. One of the most active is the German Chapter, which publishes its own newsletter in German; it is called *DISPLAY* and is edited by **Heinrich Schnepf.**

For a sample newsletter and information about the club, send a self-addressed, business-size (at least 9" long) envelope with postage attached (two 13c stamps in U.S., 2 ounces foreign postage) to:

HP-65 Users Club
Richard Nelson, Editor
2541 W. Camden Place
Santa Ana, California 92704, USA

Note: The HP-65 Users Club is not sponsored, nor in any way officially sanctioned, by Hewlett-Packard.)

ENTER ↑ 65 NOTES
EXTENDED PRECISION: A START
DECEMBER 1976 VOL 3 NO 10

NUMERICAL ANALYSIS: AN ERROR TRAP

HP-67/97 DISPLAY STROBE

OTHER SOURCES

NOP

Tips From The Users Club

These two HP-67 programming tips were recently contributed to the HP-65 Users Club newsletter, 65 NOTES, by **Art Leyenberger** of Ridgewood, New Jersey. He also sent them to us so they could be shared with HP KEY NOTES readers.

TESTING REGISTER CONTENTS OF HP-67

Using the h REG keys of the HP-67, it takes approximately 59 seconds to test all 25 registers for non-zero contents. The usual procedure is to press h REG, which will cause the display of the contents of registers 0 to 9 and 20 to 25. Then, f P≷S must be pressed and h REG pressed again to inspect registers 10 to 19 (the secondary registers). This procedure takes about 59 seconds regardless of the contents of the registers themselves.

If all that is needed is to test each register for non-zero contents, a subroutine may be used. This subroutine takes approximately 12 seconds to complete when the content of each register is zero. The maximum time for this subroutine (when all of the registers contain non-zero amounts) will be about 3 minutes.

If many registers contain non-zero values, this subroutine may be impractical. However, if only a few registers contain non-zero values and one wants to quickly locate these specific registers, this subroutine may be helpful.

To use the subroutine, press A. The display will show the register number and contents for all registers that have a non-zero value stored therein. When finished, the display will show 25.00. The subroutine is:

```
f LBL A      DSP 0
RCL (i)      f -x-
f x≠0        RCL (i)
GTO B        g x=y
f ISZ        R/S
GTO A        f -x-
f LBL B      f ISZ
h RC I       GTO A
```

TABULATOR PROGRAM TIPS

Here is a good idea to extend the application of HP-67/97 *Standard Pac* Tabulator Program, #SD-02A.

It may be desirable when creating tables to reverse the column and row parameters. By inputting the number of columns and pressing f A, instead of inputting the number of rows, after the values are entered for each row, the calculator will automatically display the row total. This may be done for each row (for as many rows as you have). Once all of the data has been entered, pressing C will result in the column totals being displayed. Naturally, when using this technique with the tabulator program, the maximum number of columns that you may have will be limited to 24.

Since the number of columns that you have is usually a fixed amount and the number of rows may vary, and when the row totals need to be calculated before being written down, I have found that it is more helpful to use this program in this manner. Also, it seems more logical to work across each row, determine the row totals,

(continued)

and then, when finished calculating, determine the column totals.

Another modification I use with this program (and other programs that I write when the size of the majority of the data to be entered is greater than 1000), is to enter four additional steps after the label A instruction. Starting with line number 28, they are: Enter, EEX, 3, and ×. This procedure allows me to enter the data as multiples of a thousand and lets the program bring the number up to its proper value. For example, the numbers 1000, 25300, 100000, and 750 would be entered as follows: 1, 25.3, 100, and .75. This saves time and reduces the amount of keypressing that I must do. It also helps me to comprehend amounts when I think of them and use them as multiples of a thousand.

OUTSMARTING THE HP-67 "SMART" CARD READER

When using either the HP-67 or HP-97 programmable calculators, one example of the "smart" card reader present in both machines is the programming aid of informing the user that there is additional information to be read from or recorded on the other side of the magnetic card (depending upon what mode the calculator is in, either W/PRGM or RUN). When recording a card, if there are more than 112 steps in the program that is currently in program memory or if there are instructions other than R/S after line 112, the display shows **Crd.** This prompts the user to insert the other side of the card into the card reader slot so that the remaining portion of the program can also be recorded. Likewise, when reading a card, if there is additional information on the other side of the card, after the first pass through the calculator the display will show **Crd** to signal the user of that fact.

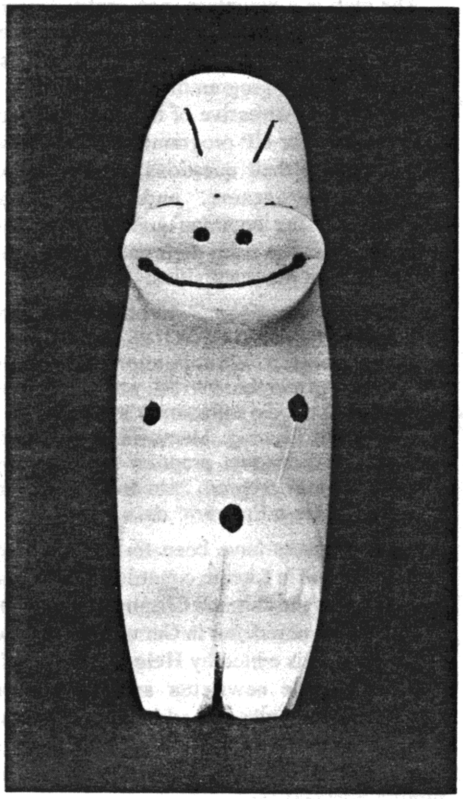
Often, the situtation arises where a program is less than 112 steps long and is recorded on side 1 of a magnetic card. Side 2 of the card may contain data that is necessary to run the program. Unfortunately, if this is the case, when side 1 of the card is read into the calculator, the display will show "0.00" (no prompting occurs). By not seeing **Crd** in the display, the user may forget to read-in side 2. Using the convention of recording a program on side 1 and the data on side 2 of a magnetic card, it would be quite helpful if the card reader would prompt for side 2 when reading a card.

Suppose a program is 90 steps in length and is in the program memory. In the W/PRGM mode, press GTO · 1 1 3, then CL x. Now record the program onto side 1 of a magnetic card. The calculator will then prompt with **Crd** in the display. Switch to the RUN mode and press any key to clear the display. Assuming the proper data is stored in the registers, R(0) thru R(9) and R(A) thru R(E), press W/DATA and pass side 2 of the card through the calculator to record the data onto side 2. Now, once the card is clipped on both sides to protect it, whenever side 1 is read into the calculator, the display will show **Crd** to prompt the user to enter side 2 (which has the data recorded on it).

Although the prompting function of the HP-67/97 is a very useful feature indeed, with the above described technique it becomes even more useful, particularly with non-programming individuals or those who may be unfamiliar with the calculator. This technique proves that the "smart" card reader is in fact smart, but not as smart as the programmer.

What's A Billiken?

The intense desire to acquire one of our new programmable calculators sometimes produces humorous situations. An example is the following letter—and photo. Will this produce results at the factory? Well, don't count on it. Orders are filled in the sequence received, and all of that depends on availability, the number of orders extant, and the volume of production. And, because we do not just "stamp-out" HP-67's and HP-97's, sometimes you have to wait a while to get one. But, anyway, here's an amusing (and *very* clever!) letter to one of our Customer Service people.



"This little Walrus Ivory Carving is called a 'Billiken.' It's an 'Alaskan Good Luck Charm' for your desk, and every morning when you arrive at work, if you rub his belly your wishes will be looked upon with favor.

I am not saying he will grant your every wish, but just looking at his smiling face will brighten your day with sunshine. I am sending him to you as my Alaskan Emissary to expedite the shipment of my HP-97. He is my *chargé d' affaires,* my representative in Court.

Your Majesty, I beg of you not to banish my Ammbassador from your Kingdom, for he speaks for me.

In years past, we Alaskans have been treated as poor step-children in our petitions to the South-48. Our requests for services either have been ignored or placed upon the bottom of the pile so that we have become very devious in achieving equal attention.

I pray that his smiling countenance will kindle a warmth that will grow from a tiny spark into a roaring conflagration, and you will bestow upon my kneeling figure one small desire, an HP-97."

Rapid Reverse Branching Made Easy

Some HP-67/97 owners have asked us for more information about rapid reverse branching as it applies to where a program will branch, based on the negative number stored in the I-register. Well, **Art Leyenberger** of Ridgewood, New Jersey, has come up with a clever idea that should clarify this function. Let's let him tell it in his words.

With regard to the rapid reverse branching capability of the calculator, I would like to offer what I think is a useful aid in determining where a program will branch to, based on the negative number stored in the I-register. For myself, at least, this formula makes the process clearer than that described in the handbook. With this formula, it is easy to determine the line number of a reverse branch instruction simply by plugging in the values of L and (i).

$$L1 = L + (i)$$

Where:

L1 = line number to which program will branch.

L = line number of the GTO or GSB instruction.

(i) = the negative number currently stored in I.

NOTE: If, after calculation of the formula, L1 is a negative number, add 224 to it to obtain the line number to which the program will branch.

HP-67/97 Programming Tips

After you have read your *Owner's Handbook* and have mastered the *operation* of your new HP-67 or HP-97, you'll want to devise clever ways to program various functions. However, because of the nature of programming, everyone isn't likely to think of every way there is to formulate routines. So, in each issue of HP KEY NOTES, we'll try to bring you innovative tips you can use.

SAVING SPACE WITH REGISTER ARITHMETIC

In many instances, the register arithmetic of the HP-67 and HP-97 can be used to save program steps. For instance, if you needed to calculate:

$$A(x + y^2)$$

$$A(x - y^2)$$

where A is stored in R_0, x is stored in R_1, and y is stored in R_2, you could do it in the stack using the code below.

```
001 LBL A                    008 RCL 1
002 RCL 1                    009 RCL 2
003 RCL 2                    010 x²
004 x²                       011 −
005 +                        012 RCL A
006 RCL A                    013 x         A(x−y²) in x
007 x          A(x+y²) in x  014 RTN
```

GTO page 9

A SOFTWARE BARGAIN HARD TO BEAT

Here is a bargain you won't want to miss! We have on hand a limited supply of "used" HP-65 application pacs that you can buy at a greatly reduced price. By "used" we mean that they have been on dealer's shelves and were traded back to us for the new HP-67/97 pacs, or they are some of the pacs returned to us on our 15-day free trial offer. But, because of various regulations, they cannot be sold as new products; therefore, *all* are marked "used".

In case you are not familiar with all of our application pacs, we have listed the contents of each pac on the back of this sheet. Look over the list carefully. The manual, alone, is probably worth the price!!

As with any "bargain," there are a few stipulations. One, the offer is good only in the United States (including Hawaii and Alaska). Two, not only are supplies limited, but there is a limit of only 10 pacs to a customer. And three, you must allow up to 4 weeks for delivery.

Although these are all officially "used" items, *they will be covered by our same Full One-Year Warranty that covers any new calculator or software pac.* (Warranty available on request.)

So, for a rare, *bona fide* bargain, fill in the order blank below and mail it with your check or money order as soon as possible. Be sure to include any state or local taxes, and don't forget to type or print your address and zip code in the address block. Don't delay, this offer is good only as long as the supply lasts.

HEWLETT hp PACKARD

If you are claiming a tax-exempt status, please enter your tax-exempt number. ______________________

00065-67001	Math Pac 1	___
00065-67002	Math Pac 2	___
00065-67003	Surveying Pac 1	___
00065-67004	Medical Pac 1	___
00065-67005	Stat Pac 1	___
00065-67007	E.E. (Elec. Engrg.) Pac 1	___
00065-67042	Aviation Pac 1	___
00065-67044	Finance Pac 1	___
00065-67045	Navigation Pac 1	___
00065-67050	Chemical Engrg. Pac 1	___
00065-67051	Stress Analysis Pac 1	___
00065-67053	Stat Pac 2	___
00065-67056	E.E. Pac 2 (Microwave)	___

Total Application Pac ___ × $19.95 = ______

Add State or Local Taxes (× %) = ______

Total Enclosed (Check or money order payable to Hewlett-Packard) = ______

Name ______________________________

Address ______________________________

City______________________________

State ____________________ Zip __________

MAIL TO: HEWLETT-PACKARD
P.O. Box 3400
Corvallis, Oregon 97330

The HP-65 can be "tailor made" for your profession.

Pre-recorded programs let you transform your HP-65 into a specialized calculator–in seconds.

To concentrate the capabilities of the HP-65 in your field of interest, you can select from an ever-expanding variety of pre-recorded magnetic program cards packaged in Application Pacs. Each Pac contains cards for up to 40 programs, a detailed manual on the Pac, and a set of Pocket Instruction Cards for listing your program instructions.

FINANCE PAC 1

These 38 programs (on 40 pre-recorded cards) help give you fast answers to complex problems in investment analysis, loans, leasing, savings and annuities, business statistics and other business and financial applications. Included are programs for:

- COMPOUND AMOUNT
- DIRECT REDUCTION LOAN
- DIRECT REDUCTION LOAN WITH BALLOON PAYMENT
- SINKING FUND

Other programs are:

- PERIODIC SAVINGS, ANNUITY DUE
- PRESENT VALUE, ANNUITY DUE
- PRESENT VALUE, ANNUITY DUE WITH BALLOON PAYMENT
- SAVINGS-COMPOUNDING PERIODS DIFFERENT FROM PAYMENT PERIODS
- NOMINAL TO EFFECTIVE/EFFECTIVE TO NOMINAL RATE CONVERSION
- DIRECT REDUCTION LOAN; ACCUMULATED INTEREST/REMAINING BALANCE
- DIRECT REDUCTION LOAN; AMORTIZATION SCHEDULE
- ADD-ON RATE INSTALLMENT LOAN
- CONSTANT PAYMENT TO PRINCIPAL LOAN AMORTIZATION SCHEDULE
- INTEREST REBATE—RULE OF 78'S
- INTERNAL RATE OF RETURN, UNEVEN CASH FLOWS
- DISCOUNTED CASH FLOW ANALYSIS, NET PRESENT VALUE
- STRAIGHT LINE DEPRECIATION SCHEDULE
- SUM-OF-THE YEAR'S DIGITS DEPRECIATION SCHEDULE
- VARIABLE RATE DECLINING BALANCE DEPRECIATION SCHEDULE
- CROSSOVER POINT-DECLINING BALANCE TO STRAIGHT LINE
- DAYS BETWEEN DATES
- BOND PRICE AND YIELD
- ACCRUED SIMPLE INTEREST
- LINEAR REGRESSION (Trend Line)
- EXPONENTIAL CURVE FIT (Growth Curve)
- TOTAL, AVERAGE AND PERCENT OF TOTAL
- MOVING AVERAGES
- INVOICING

00065-67044 . . . . $45.00

MATH PAC 1

This basic series of 40 programs (on 40 pre-recorded cards) speeds the handling of problems in algebra, trigonometry, analytic geometry and calculus. It covers:

- HYPERBOLIC FUNCTIONS
- FIRST ORDER DIFFERENTIAL EQUATION
- SOLUTION OF A TRIANGLE (Given a, b, c or a, b, C)
- 3 x 3 MATRIX INVERSION
- FIFTH DEGREE POLYNOMIAL EQUATION
- FACTORS OF AN INTEGER

Other programs are:

- GREATEST COMMON DIVISOR, LEAST COMMON MULTIPLE
- ARITHMETIC AND HARMONIC PROGRESSIONS
- GEOMETRIC PROGRESSION
- FUNCTIONS OF x AND y
- QUADRATIC EQUATION
- CUBIC EQUATION
- FOURTH DEGREE POLYNOMIAL EQUATION
- SIMULTANEOUS EQUATIONS IN TWO UNKNOWNS
- SIMULTANEOUS EQUATIONS IN THREE UNKNOWNS
- SYNTHETIC DIVISION
- RECTANGULAR, SPHERICAL CONVERSIONS
- TRANSLATION AND/OR ROTATION OF COORDINATE AXES
- ANGLE CONVERSIONS
- SECONDARY VALUES OF $\sin^{-1}$, $\cos^{-1}$, $\tan^{-1}$
- TRIGONOMETRIC FUNCTIONS
- INVERSE HYPERBOLIC FUNCTIONS
- SOLUTION OF A TRIANGLE (GIVEN a, A, C or a, B, C)
- SOLUTION OF A TRIANGLE (GIVEN B, b, c)
- SPHERICAL TRIANGLES
- AREA OF A TRIANGLE
- AREA OF A POLYGON
- CIRCLE DETERMINED BY THREE POINTS
- EQUALLY SPACED POINTS ON A CIRCLE
- POLYGONS INSCRIBED IN AND CIRCUMSCRIBED ABOUT A CIRCLE
- UNIT CONVERSIONS: C →F; ft, in → cm; lb →kg
- UNIT CONVERSIONS: mi →km; gal → ltr; yd →m; ac →ft²
- POLYNOMIAL EVALUATION (Real)
- LINEAR AND LAGRANGIAN INTERPOLATIONS
- FINITE DIFFERENCE INTERPOLATION
- NUMERICAL INTEGRATION (Discrete Case)
- SIMPSON'S RULE FOR NUMERICAL INTEGRATION
- ROOTS OF $f(X)=0$ IN AN INTERVAL
- DETERMINANT AND CHARACTERISTIC EQUATION OF A 3 x 3 MATRIX
- 2 x 2 MATRIX OPERATIONS

00065-67001 . . . . $45.00

MATH PAC 2

A variety of advanced mathematical functions is provided by the 37 programs (40 pre-recorded cards) in this Pac. The programs are:

- BASE CONVERSION
- COMPLEX FUNCTIONS $|z|$, z, z, $1/z$
- GAUSSIAN QUADRATURE FOR $\int_a^b f(x)dx$
- BESSEL FUNCTION $J_n(x)$
- COMPLETE ELLIPTIC INTEGRALS

Other programs are:

- COMPLEX ARITHMETIC
- COMPLEX TRIGONOMETRIC AND HYPERBOLIC FUNCTIONS
- COMPLEX INVERSE TRIGONOMETRIC AND HYPERBOLIC FUNCTIONS
- OCTAL ARITHMETIC
- INTEGER BASE CONVERSION
- COMPLEX FUNCTIONS z^n, $z^{1/n}$
- COMPLEX FUNCTIONS e^z, $\ln z$, a^z, $\log_a z$
- COMPLEX FUNCTIONS z^w, $z^{1/w}$, $\log_z w$
- POLYNOMIAL EVALUATION (Complex)
- INTERSECTIONS OF A LINE AND A CONIC SECTION
- VECTOR PRODUCTS AND ANGLE BETWEEN VECTORS
- PARTIAL SUM AND PARTIAL PRODUCT
- GAUSSIAN QUADRATURE FOR $\int_a^\infty f(x)dx$
- KELVIN FUNCTIONS
- EULER Φ FUNCTION
- GAMMA FUNCTION
- INCOMPLETE GAMMA FUNCTION
- ERROR FUNCTION AND COMPLEMENTARY ERROR FUNCTION
- CONFLUENT HYPERGEOMETRIC FUNCTION
- GAUSSIAN HYPERGEOMETRIC FUNCTION
- CHEBYSHEV POLYNOMIAL
- LEGENDRE POLYNOMIAL
- HERMITE POLYNOMIAL
- LAGUERRE POLYNOMIAL
- SINE INTEGRAL
- COSINE INTEGRAL
- EXPONENTIAL INTEGRAL
- FRESNEL INTEGRALS

00065-67002 . . . . $45.00

STAT PAC 1

Basic and advanced functions—37 programs (40 pre-recorded cards)—selected from the areas of general statistics; distribution functions; curve fittings; and test statistics. Programs included in this Pac are:

- MEAN, STANDARD DEVIATION, STANDARD ERROR
- RANDOM NUMBER GENERATOR
- ANALYSIS OF VARIANCE (One Way)
- MULTIPLE LINEAR REGRESSION
- 2 x k CONTINGENCY TABLE
- SUMS FOR TWO VARIABLES
- BASIC STATISTICS (Two Variables)

Other programs are:

- MEAN, STANDARD DEVIATION, STANDARD ERROR (Grouped Data)
- PERMUTATION AND COMBINATION ARITHMETIC, GEOMETRIC, HARMONIC AND GENERALIZED MEANS
- MOMENTS, SKEWNESS AND KURTOSIS (Grouped or Ungrouped Data)
- NORMAL DISTRIBUTION
- INVERSE NORMAL INTEGRAL
- CHI-SQUARE DISTRIBUTION
- t DISTRIBUTION
- F DISTRIBUTION
- BIVARIATE NORMAL DISTRIBUTION
- LOGARITHMIC NORMAL DISTRIBUTION
- WEIBULL DISTRIBUTION
- BINOMIAL DISTRIBUTION
- NEGATIVE BINOMIAL DISTRIBUTION
- HYPERGEOMETRIC DISTRIBUTION
- POISSON DISTRIBUTION
- LINEAR REGRESSION
- EXPONENTIAL CURVE FIT
- POWER CURVE FIT
- LOGARITHMIC CURVE FIT
- LEAST SQUARES REGRESSION OF $y = cx^a + dx^b$
- PARABOLIC CURVE FIT
- PAIRED t STATISTIC
- t STATISTIC FOR TWO MEANS
- CHI-SQUARE EVALUATION
- BARTLETT'S CHI-SQUARE STATISTIC
- SPEARMAN'S RANK CORRELATION COEFFICIENT
- MANN-WHITNEY STATISTIC
- KENDALL'S COEFFICIENT OF CONCORDANCE
- BISERIAL CORRELATION COEFFICIENT

00065-67005 . . . . $45.00

STAT PAC 2

Thirty-one programs (40 pre-recorded cards) provide solutions in the areas of general statistics, distribution functions, curve fitting, analysis of variance, test statistics, probability, quality control, and queing theory.

- PARTIAL AND MULTIPLE CORRELATION COEFFICIENTS
- MOVING AVERAGES (Order 2 to 8)
- HISTOGRAM (12 Intervals)
- F DISTRIBUTION WITH ODD DEGREES OF FREEDOM
- ERLANG DISTRIBUTION (Gamma Distribution)
- GEOMETRIC CURVE FIT
- GOMPERTZ CURVE FIT
- WEIBULL DISTRIBUTION PARAMETER CALCULATION
- WEIGHTED REGRESSION (Special Case)
- POLYNOMIAL APPROXIMATION
- TWO WAY ANALYSIS OF VARIANCE (No Replications)
- TWO WAY ANALYSIS OF VARIANCE (With Replications)
- LATIN SQUARE
- ANALYSIS OF COVARIANCE (One Way)
- ONE SAMPLE TEST STATISTICS FOR THE MEAN
- TEST STATISTICS FOR THE CORRELATION COEFFICIENT
- DIFFERENCES AMONG PROPORTIONS
- BEHRENS-FISHER STATISTIC
- KRUSKAL-WALLIS STATISTIC
- MEAN-SQUARE SUCCESSIVE DIFFERENCE
- 3 x K CONTINGENCY TABLE
- THE RUN TEST FOR RANDOMNESS
- INTRACLASS CORRELATION COEFFICIENT
- FISHER'S EXACT TEST FOR A 2 x 2 CONTINGENCY TABLE
- PROBABILITY OF NO REPETITIONS IN A SAMPLE (Birthday Problem)
- $\bar{x}$ and R CONTROL CHARTS
- p AND c CONTROL CHARTS
- OPERATING CHARACTERISTIC CURVE (Type A)
- OPERATING CHARACTERISTIC CURVE (Type B)
- SINGLE- AND MULTI-SERVER QUEUES (Infinite Customers)
- SINGLE- AND MULTI-SERVER QUEUES (Finite Customers)

00065-67053 . . . . $45.00

EE PAC 1

Thirty-five programs (40 pre-recorded cards) provide functions for use in electronic and electrical engineering projects. Topics addressed include impedance matching filter design; transmission line calculation; parameter conversion; power supply design; transistor biasing; control system and waveform analyses. The programs are:

- REACTANCE CHART
- IMPEDANCE OF LADDER NETWORK
- TRANSMISSION LINE IMPEDANCE TRANSFORMATION
- S⇄Y PARAMETER CONVERSION
- FOURIER SERIES

Other programs are:

- SERIES RESONANT CIRCUIT
- PARALLEL RESONANT CIRCUIT
- T ATTENUATOR
- PI ATTENUATOR
- WYE-DELTA OR DELTA-WYE
- TRANSFORMATION
- MINIMUM-LOSS PAD MATCHING
- PI NETWORK IMPEDANCE MATCHING
- BAND PASS FILTER DESIGN
- ACTIVE FILTER—LOW PASS
- ACTIVE FILTER—HIGH PASS
- BUTTERWORTH FILTER
- CHEBYSHEV FILTER
- CAPACITANCE OF PARALLEL PLATES
- SELF-INDUCTANCE OF STRAIGHT ROUND WIRE
- INDUCTANCE OF A SINGLE-LAYER CLOSE-WOUND COIL
- SKIN EFFECT AND COIL Q
- TRANSFORMER DESIGN
- REED RELAY DESIGN
- IMPEDANCE OF TRANSMISSION LINE
- MICROSTRIP TRANSMISSION LINE
- POWER SUPPLY RECTIFIER CIRCUITS
- CONTROLLED RECTIFIER CIRCUITS
- INTEGRATED CIRCUIT CURRENT SOURCE
- TRANSISTOR BIAS
- JFET BIAS AND TRANSCONDUCTANCE
- PHASE-LOCKED LOOP
- DECIBEL CONVERSION
- VOLTAGE TO dBm
- WIRE TABLES AI & ANNEALED Cu
- HEAT SINKS

00065-67007 . . . . $45.00

EE PAC 2 (Microwave)

Twenty-seven programs (40 pre-recorded cards) will assist the microwave circuit designer in making microwave measurements, designing transistor amplifiers, computing transmission line properties and certain system properties, and performing difficult related mathematical operations.

- MISMATCH ERROR LIMITS
- MULTIPLE MISMATCH ERROR LIMITS
- SMITH CHART: RADIALLY SCALED PARAMETERS
- SMITH CHART: IMPEDANCE ⇄ REFLECTION COEFFICIENT
- MICROSTRIP CALCULATIONS (three cards)
- TRANSMISSION LINE CALCULATIONS (three cards)
- CUTOFF FREQUENCY IN COAX
- RECTANGULAR WAVEGUIDE CALCULATIONS
- FREQUENCY CONVERSIONS
- PULSE SPECTRUM ANALYSIS
- SPURIOUS RESPONSES
- FM SIDEBAND LEVEL
- MODULATION INDEX FOR SPECIFIED CARRIER SUPPRESSION
- CONSTANT-EXCESS NOISE MEASUREMENT
- NOISE FIGURE OF CASCADED NETWORKS
- IMPEDANCE MATCHING
- UNILATERAL DESIGN: FIGURE OF MERIT, MAXIMUM UNILATERAL GAIN
- UNILATERAL DESIGN: GAIN CIRCLES
- UNILATERAL DESIGN: NOISE FIGURE CIRCLES
- BILATARAL DESIGN: STABILITY FACTOR, MAXIMUM GAIN, OPTIMUM MATCHING (three cards)
- BILATERAL DESIGN: GAIN CIRCLES
- BILATERAL DESIGN: STABILITY CIRCLES
- LOAD AND SOURCE MAPPING
- LINEAR AND LAGRANGIAN INTERPOLATION
- PARAMETER CONVERSION: S⇄Y, Z, G, H (six cards)
- PARAMETER CONVERSION: S⇄T (two cards)
- COMPLEX MATRIX OPERATIONS (two cards)

00065-67056 . . . . $45.00

CHEMICAL ENGINEERING PAC 1 (Thermal and Transport Science)

18 programs (on pre-recorded cards) designed to aid the engineer in thermodynamic and transport process calculations. Topics addressed inclue P-V-T gas relations, gas dynamics, incompressible flow, heat exchangers, heat conduction, black body radiation, and curve fitting.

- IDEAL GAS EQUATION OF STATE
- REDLICH—KWONG EQUATION OF STATE
- REVERSIBLE POLYTROPIC PROCESS FOR AN IDEAL GAS
- ISENTROPIC FLOW FOR IDEAL GASES
- ONE DIMENSIONAL NORMAL SHOCKS FOR IDEAL GASES

Other programs are:

- FLUID TRANSPORT NUMBERS (NUSSELT—HEAT & MASS, REYNOLDS, STANTON, LEWIS, SCHMIDT, BIOT, AND PRANDTL)
- FANNING FRICTION FACTOR AND CONDUIT FLOW
- CONSERVATION OF ENERGY
- VON KÁRMÁN ANALOGY FOR HEAT AND MASS TRANSFER
- HEAT EXCHANGER ANALYSIS (HEAT TRANSFER AND EFFECTIVENESS FOR CROSS-FLOW, COUNTER-FLOW, PARALLEL-FLOW AND PARALLEL-COUNTER-FLOW HEAT EXCHANGERS)
- HEAT TRANSFER THROUGH COMPOSITE CYLINDERS AND WALLS
- STRAIGHT FIN EFFICIENCY
- NATURAL CONVECTION (ESTIMATE COEFFICIENTS FOR VERTICAL CYLINDERS AND WALLS, AND HORIZONTAL CYLINDERS AND PLATES)
- BLACK BODY THERMAL RADIATION TEMPERATURE OR CONCENTRATION PROFILE FOR A SEMI-INFINITE SOLID
- HYDROCARBON COMBUSTION
- CURVE FITTING (LINEAR, EXPONENTIAL, AND POWER)
- UNIT CONVERSIONS

00065-67050 . . . . $45.00

STRESS ANALYSIS PAC 1

25 programs (on 39 pre-recorded cards) designed to aid the engineer in the calculation of the properties of structural elements. Topics addressed include vector statics, section properties, interference fits, stress analysis, flat plates, beams and columns. Programs included in this Pac are:

- TWO DIMENSIONAL VECTOR OPERATIONS
- STATIC EQUILIBRIUM OF A POINT
- STATIC EQUILIBRIUM OF A RIGID BODY

Other programs are:

- PROPERTIES OF RECTANGULAR SECTIONS
- PROPERTIES OF CIRCULAR SECTIONS
- PROPERTIES OF ANNULAR SECTIONS
- COMPOSITE SECTION PROPERTIES
- BENDING STRESS IN BEAMS OR TORSIONAL SHEAR STRESS IN CIRCULAR SHAFTS
- LINEAR OR ANGULAR DEFORMATION OF A SHAFT
- THIN-WALLED PRESSURE VESSELS
- STRESS IN THICK-WALLED CYLINDERS
- INTERFERENCE FITS
- MOHR CIRCLE FOR STRESS
- SODERBERG'S EQUATION FOR FATIGUE
- CIRCULAR PLATES WITH SIMPLY SUPPORTED EDGES
- CIRCULAR PLATES WITH FIXED EDGES
- RECTANGULAR PLATES (SIMPLY SUPPORTED AND WITH FIXED EDGES)
- COMPRESSIVE BUCKLING
- ECCENTRICALLY LOADED COLUMNS
- RECTANGULAR, REINFORCED CONCRETE SECTIONS
- BOLT TORQUE

The next four programs calculate deflection, slope, moment, and shear for various beam geometries.

- CANTILEVER BEAMS
- SIMPLY SUPPORTED BEAMS
- BEAMS FIXED AT BOTH ENDS
- BEAMS FIXED AT ONE END AND SIMPLY SUPPORTED AT THE OTHER

00065-67051 . . . . $45.00

SURVEYING PAC 1

Because surveying practices differ from country to country, different versions are offered. Contact your local Hewlett-Packard sales office for the exact contents of Surveying Pac 1 in your country. The U.S. version consists of 34 programs (on 34 pre-recorded cards) dealing with traversing; curves; triangles and intersections; pre-determined area; and earthwork. The Pac contains:

- FIELD ANGLE TRAVERSE
- HORIZONTAL CURVE LAYOUT
- BEARING-DISTANCE INTERSECT
- STADIA REDUCTIONS
- VOLUME BY AVERAGE END AREA

Other programs are:

- BEARING TRAVERSE
- CLOSURE FOR FIELD ANGLE AND BEARING TRAVERSES
- INVERSE FROM COORDINATES
- SIDESHOTS
- COORDINATE TRANSFORMATION
- COMPASS RULE ADJUSTMENT
- TRANSIT RULE ADJUSTMENT
- TWO INSTRUMENT RADIAL SURVEY
- CURVE SOLUTION—Given △& R or △& T
- CURVE SOLUTION—Given R & T or R & L
- CURVE SOLUTION—Given △& C or R & C
- ELEVATIONS ALONG A VERTICAL CURVE
- TRIANGLE SOLUTION—Given SSS or SAS
- TRIANGLE SOLUTION—Given SSA
- TRIANGLE SOLUTION—Given ASA or AAS
- BEARING-BEARING INTERSECT
- DISTANCE-DISTANCE INTERSECT
- DISTANCE FROM A POINT TO A LINE
- TAPING CORRECTIONS
- EDM SLOPE REDUCTION—Given Zenith Angle
- EDM SLOPE REDUCTION—Given △ Elevation
- FIELD ANGLE CHECK
- THREE WIRE LEVELING
- SLOPE STAKING—Given Centerline Terrain Elevation
- SLOPE STAKING—Given Centerline Cut/Fill
- AZIMUTH OF THE SUN
- PREDETERMINED AREA—Line Through a Point
- PREDETERMINED AREA—Two Sides Parallel
- VOLUME OF BORROW PIT

00065-67003 . . . . $45.00

MACHINE DESIGN PAC 1

Thirty-five programs (40 pre-recorded cards) provide solutions for the machine designer in dynamics, vibrations, linkages, cams, gears, springs, power transmission, and machine geometries.

- CONSTANT ACCELERATION—TIME
- CONSTANT ACCELERATION—VELOCITY
- KINETIC ENERGY
- FREE VIBRATIONS
- VIBRATION FORCED BY $F_0 \cos\omega t$
- FORCED OSCILLATOR WITH ARBITARY FUNCTION
- FOURIER SERIES
- CRITICAL SHAFT SPEED
- FOUR BAR FUNCTION GENERATOR
- PROGRESSION OF FOUR BAR SYSTEM
- LINEAR PROGRESSION OF SLIDER CRANK
- ANGULAR PROGRESSION OF SLIDER CRANK
- CAM DATA STORAGE
- HARMONIC CAM DESIGN—RADIAL ROLLER FOLLOWER
- HARMONIC CAM DESIGN—FLAT FACED FOLLOWER
- ROLLER FOLLOWER CAM FUNCTION GENERATOR
- FLAT FACED FOLLOWER CAM FUNCTION GENERATOR
- LINEAR CAM FUNCTION GENERATOR
- SPUR GEAR REDUCTION DRIVE
- STANDARD EXTERNAL INVOLUTE SPUR GEARS
- SPUR/HELICAL GEAR FORCES
- BEVEL GEAR FORCES
- WORM GEAR FORCES
- SPRING CONSTANT
- HELICAL SPRING DESIGN
- TORSION SPRING DESIGN
- FLAT SPRING DESIGN
- CONE AND PLATE CLUTCHES
- POWER SCREWS
- RPM/TORQUE/POWER
- LINE—LINE INTERSECTION/GRID POINTS
- CIRCLE—LINE INTERSECTION
- CIRCLE—CIRCLE INTERSECTION
- POINTS ON A CIRCLE
- BELT LENGTH

00065-67052 . . . . $45.00

MEDICAL PAC 1

This Pac of 27 programs (38 pre-recorded cards) allows for the following types of medical calculations: unit conversions; ventilator set-up and calibration; analysis of cardio-pulmonary function; acid-base balance, blood gases; and respiratory status. The Pac contains:

- WEIGHT CONVERSIONS
- LENGTH CONVERSIONS
- VOLUME CONVERSIONS
- ENGLISH-METRIC CONVERSIONS
- PATIENT IDENTIFICATION

Other programs are:

- MALE PULMONARY FUNCTIONS
- FEMALE PULMONARY FUNCTIONS
- LUNG DIFFUSION
- RESPIRATORY GAS CONVERSIONS
- VENTILATOR SETUP
- $PaCO_2$ NORMALIZATION
- BLOOD ACID-BASE STATUS
- VIRTUAL PO_2
- OXYGEN SATURATION AND CONTENT
- ANAEROBIC PCO_2 AND pH CHANGE
- ANAEROBIC PO_2 CHANGE
- DEAD SPACE FRACTION
- A-a O_2 DIFFERENCE
- PHYSIOLOGIC SHUNT AND FICK
- DUBOIS BODY SURFACE AREA
- BOYD BODY SURFACE AREA
- DYE CARDIAC OUTPUT
- FICK CARDIAC OUTPUT
- VALVE AREA
- ANATOMIC SHUNTS
- CONTRACTILITY
- STROKE WORK

00065-67004 . . . . $45.00

AVIATION PAC 1

Here are 29 programs (31 cards) for flight and preflight calculations—primarily for the private and business pilot, but also being used by many airline pilots. Included are:

- AIRCRAFT FLIGHT PLAN WITH WIND
- FLIGHT MANAGEMENT
- PREDICTING FREEZING LEVELS
- GENERAL AIRCRAFT WEIGHT AND BALANCE
- CUSTOMIZED WEIGHT AND BALANCE
- TURN PERFORMANCE

Other programs are:

- RATE OF CLIMB AND DESCENT
- HEAD WINDS AND CROSSWINDS
- FLIGHT PLANNING AND FLIGHT VERIFICATION
- DETERMINING IN-FLIGHT WINDS
- STANDARD ATMOSPHERE (For altitudes 0-36089 and 36089-82000)
- MACH NUMBER AND TRUE AIRSPEED
- TRUE AIR TEMPERATURE AND DENSITY ALTITUDE
- LOWEST USABLE FLIGHT LEVEL
- GREAT CIRCLE PLOTTING
- RHUMBLINE NAVIGATION
- GREAT CIRCLE NAVIGATION
- POSITION, GIVEN HEADING, SPEED AND TIME
- LINE OF SIGHT DISTANCE
- POSITION BY TWO VORS
- NAVIGATION BY TWO VORS
- POSITION BY ONE VOR
- DME SPEED CORRECTION
- AVERAGE WIND VECTOR
- COURSE CORRECTION
- TIME OF SUNRISE AND SUNSET
- AZIMUTH OF SUNRISE AND SUNSET
- PILOT UNIT CONVERSIONS
- CUSTOMIZED UNIT CONVERSIONS

00065-67042 . . . . $45.00

NAVIGATION PAC 1

Written for marine navigators but also useful to the land or air navigator, these 26 programs (40 cards) assist the navigator in piloting, dead reckoning, celestial navigation, and relative motion problems. The Pac includes:

- GREAT CIRCLE COMPUTATION
- LONG-TERM ARIES ALMANAC
- 1974-1975 SUN ALMANAC
- LONG-TERM STAR ALMANAC
- ALMANAC POSITIONS
- SIGHT REDUCTION TABLE
- MANEUVERING RELATIVE TO ANOTHER VESSEL

Other programs are:

- LENGTH CONVERSIONS
- SPEED, TIME, AND DISTANCE
- TIME-ARC CONVERSION
- PROPELLER SLIP
- FUEL CONSUMPTION
- DISTANCE TO OR BEYOND HORIZON
- DISTANCE BY HORIZON ANGLE AND DISTANCE SHORT OF HORIZON
- DEAD RECKONING
- RHUMBLINE NAVIGATION
- GREAT CIRCLE NAVIGATION
- COMPOSITE SAILING
- SEXTANT ALTITUDE CORRECTIONS
- SUNRISE, SUNSET, AND TWILIGHT
- MOST PROBABLE POSITION
- FIX BY TWO OBSERVATIONS
- FIX BY THREE OBSERVATIONS
- DISTANCE OF AN OBJECT BY TWO BEARINGS
- VECTOR ADDITION
- VELOCITY TO CHANGE RELATIVE POSITION

00065-67045 . . . . $45.00

HEWLETT hp PACKARD

Or, you could save steps by calculating $A(x+y^2)$ in the stack and $A(x-y^2)$ in R_1, as follows:

```
001 LBL A                        007 RCL A
002 RCL 1                        008 STO×1   A(x−y²)in R₁
003 RCL 2                        009 x       A(x+y²)in x
004 x²                           010 RCL 1   A(x−y²)in x
005 STO−1     (x−y²)in R₁        011 RTN
006 +         (x+y²)in x
```

This is a code savings of 21%.

ARC TANGENT OF (x/y)

In many applications the arc tangent of the ratio of two quantities (x/y) must be calculated. This is a trivial problem unless the value of y is zero. In this case, the division of x by y would cause an **Error,** and program execution would halt.

One way to overcome this problem, in any angular mode, is to use the following subroutine instead of ÷, $\tan^{-1}$.

```
LBL 1
→P        Calculate θ(−180 ≤ θ ≤ 180).
R↓        Move θ to x.
cos    }  Test θ to determine if it is in 2nd or 3rd
x<0    }  quadrants.
GTO 0  }
CLx    }  If θ is in 1st or 4th quadrant, return final
LST x  }  value of θ.
RTN    }
LBL 0  }
CLx    }  Return 2nd or 3rd quadrant θ to x.
LST x  }
1      }  Compute and add 180°, π rad, or 200
CHS    }  grads, depending on angular mode, to
cos⁻¹  }  bring angle back to 1st or 4th quadrant.
+      }
1      }  Convert angles to standard arc tangent
→R     }  notation of (−90 ≤ θ ≤ 90) instead of
→P     }  (0 ≤ θ ≤ 90) and (270 ≤ θ ≤ 360).
R↓     }
RTN
```

THE PRINT/PAUSE OR STOP ROUTINE

The following routine (when called as a subroutine) automatically causes a print/pause of multiple results if the print flag is on, or it permits you to press **R/S** to obtain each result. See page L08-01 of the HP-67/97 *Standard Pac* for a description of setting and clearing the print flag.

```
LBL 9       RTN
F0?         LBL 1
GTO 1       PRINT x
R/S         RTN
```

When subroutine 9 is called (GSB 9), flag 0 is tested. If the flag is on, the calculator moves to LBL 1, prints/pauses the display, and then returns to the main program. If flag 0 is off, the calculator stops at R/S. When **R/S** is pressed from the keyboard, the calculator executes the RTN command, goes back to the main program, and continues execution.

Reversing The "Sense" Of Test-Cleared Flags F2 & F3

We received the following excellent idea and presentation from **Chris Johansen** of Auburndale, Massachusetts. If you are having trouble understanding or applying flags, here is an easy—and very clear—application you can study and perhaps use for your programs.

Occasionally, a user of the HP-67/97 would like his/her program to perform a single keystroke if a condition is false. While it is sometimes possible to design the program so that a true flag indicates the condition is false and a false flag indicates the true condition, it is more easily recalled and, for instance, in the case of numeric entry indicated by flag F3, it is sometimes demanded that the flag and the condition it indicates have the same logical sense.

As a contrived example, suppose a user wants to write an endless-loop program (or segment) that first presents a "cue" number (7, in this case) to request a possible input, then adds to register R_1 either the negative of the user's input or a zero, and then displays the current sum. Subroutines A and B, below, show two approaches. Shorter subroutine C takes advantage of the fact that the negative of zero is also zero. Subroutine D illustrates the main point of this letter: If there has been a data entry, F3 is true, and the first F3? clears F3 and performs the very next step, F3?, which, since F3 is clear (false) must skip over the following CLX; if there *wasn't* any data entry, F3 is already clear, and the first F3? skips a step to the CLX. The effect is to reverse the "DO if TRUE" rule, only clearing X if F3 is false. Finally, subroutine E proves that the shortest program may need to use a different approach.

```
002  *LBLA    018  *LBLB    034  *LBLC    048  *LBLD    061  *LBLE
003      7    019      7    035      7    049      7    062      7
004    PSE    020    PSE    036    PSE    050    PSE    063    PSE
005    F3?    021    F3?    037    F3?    051    F3?    064    F3?
006   GTO2    022   GTO2    038   GTO2    052    F3?    065   ST-1
007    CLX    023    CLX    039    CLX    053    CLX    066   RCL1
008  *LBL1    024   GTO1    040  *LBL2    054    CHS    067    PSE
009   ST+1    025  *LBL2    041    CHS    055   ST+1    068   GTOE
010   RCL1    026    CHS    042   ST+1    056   RCL1
011    PSE    027  *LBL1    043   RCL1    057    PSE
012   GTOA    028   ST+1    044    PSE    058   GTOD
013  *LBL2    029   RCL1    045   GTOC
014    CHS    030    PSE
015   GTO1    031   GTOE
```

The "Void" Maker

Here is a note from **Tak Y. Lee** of Wellesley, Massachusetts. Not everyone has an HP-97, so we added the tape printouts to clarify his idea. When you roll-down the entries in the stack registers, you see only one at a time. The HP-97's tape graphically (and convincingly) shows what happens to the stack registers when you use **CLx** and **R↓** to clear the X-register.

CLx and **R↓** seem to be two interchangeable keys for clearing the X-register to make way for further computations. In many instances it would seem that **CLx** is preferable to **R↓**, especially when you like to duplicate the contents in the T-register every time you drop the stack. But what you don't want is a "void" in your stack-loop. Don't jump to conclusions until you try the following programs (A & B) and see what key really is the "void" maker.

```
LBL A       LBL B       LBL 1
4           4           1
ENTER       ENTER       RTN
ENTER       ENTER
ENTER       ENTER
CLX         R↓
GSB 1       GSB 1
RTN         RTN
```

The results *look* all right. But, wait! Print out the stack and compare again.

So, **CLx**, after all, *is* the key that puts a "void" in your stack-loop! As you can see, **CLx**, if followed by such keys as **LBL**, **GTO**, **GSB**, **P⇄S**, etc., causes the stack to subsequently "lift" on data entry, and you get the result shown:

```
           GSBA                    GSBB
001       *LBLA        009        *LBLB
002           4        010            4
003        ENT↑        011         ENT↑
004        ENT↑        012         ENT↑
005        ENT↑        013         ENT↑
006         CLX        014           R↓
 0.00       ***         4.00        ***
007        GSB1        015         GSB1
017       *LBL1        017        *LBL1
018           1        018            1
019         RTN        019          RTN
008         RTN        016          RTN

           PRST                    PRST

 4.00         T         4.00          T
 4.00         Z         4.00          Z
 0.00         Y         4.00          Y
 1.00         X         1.00          X
```

Application Pac Corrections

If you own some of our application pacs, check the following corrections and mark them in your copy—or mail in your old card and we will send you a revised card. If your copy is correct, you have a later, revised issue of the book and/or card.

HP-67/97 MATH PAC 1

The program on the magnetic card for *Optimal Scale, Plotting,* MA1-04A, contains four more steps than the program listing on pages L04-01 and L04-02; the prerecorded card, however, is correct. Change your book as follows:

On page 04-02, the first sentence of the first paragraph should start with 92, not 96. On the same page, change both "2. Press GTO • 1 2 8." steps to: "2. Press GTO • 1 3 2."

On page 04-03, change 128 in step 7 to 132.

On page 04-05, the keystrokes for Example 3 should be: GTO • 1 3 2.

The labels that have been changed are:

```
021  LBL 9      071  LBL D
 .   ISZ I       .   x<0?
 .   RCL E       .   GTO 0
 .   RCL A       .   INT
 .   ÷           .   RTN
 .   GSB D       .   LBL 0
 .   RCL A       .   ↑
 .   x           .   INT
029  STO 9       .   x=y?
                 .   RTN
                 .   1
                 .   –
                083  RTN
```

HP-67/97 E.E. PAC 1

The magnetic card for EE1-09A1, *Butterworth or Chebyshev Filter Design,* is recorded correctly, but the listing on page L09-01 is one step short. There should be a STO E at step 97; thus, everything from there to the end is pushed down one step.

HP-67/97 SURVEYING PAC 1

A few of the original prerecorded cards for program SU1-14A, *Predetermined Area,* and the corresponding listing are incorrect. To determine if you have an incorrect card, run Example 1 on page 14-04 (under the heading "Two Sides Parellel"). If running the example results in an **Error** display after the area 36000.0000 is displayed, you have an incorrect card. A corrected card can be obtained by sending your old card to *Customer Support* (U.S. address on back cover).

Meanwhile, mark the following changes in your book, on pages L14-01 and L14-02.

Line Number	Change
058	Delete SPC
115	Change R/S to GTO 3
Between 136 and 137	Insert LBL 3
196	Change STO 5 to STO C
201	Change STO 2 to STO B

To check for another possible error, load data card SU1-17A3 (both sides) and press f P⇄S RCL 9. The number displayed should be: 1047.546710. (We have found some cards with: 1048.546710.) If your card is incorrect, send it back to us (see above) and we will mail back a correct card.

If You Like Statistics You'll Love 00755A

Sometimes we receive a program that deserves to be highlighted in HP KEY NOTES. This one truly fits that category. Don't let the old number fool you; this is a complete revision of the original 00755. It is, in fact, a totally different program ... and *two programs* at that! So, if you are a statistically minded owner of the HP-65, you'll definitely want to order this contribution to the science of statistics.

The author is a professor of Management Science at the Tulane University Graduate School of Business Administration. And, we might add, a very astute HP-65 programmer. **Dr. Beckwith** has contributed 40 programs in the HP-65 Users' Library. Here's the abstract for #00755A.

Edgeworth Series: Frequency Function and Probability Integral (#00755A)*

Card 1 "fits" an Edgeworth Series curve (Edgeworth's form of the Gram-Charlier Series of Type A), based on the first four population or sample moments, and delivers the ordinate of the resulting frequency function. Card 2 delivers the ordinate of the corresponding distribution function, or probability integral. Provision is made for the automatic entry of (sample) parameter data following the use of the "Four Moments" program. (Catalog number 00754A, Four Moments, Skewness, Excess, Kurtosis—also one of Dr. Beckwith's programs.)

Author: **Dr. Richard E. Beckwith**
New Orleans, Lousiana

In European areas, order by number 51519A.

One More Time!

We have published many routines on the HP-65's firmware conversions, f→D.MS and f⁻¹→D.MS, because the HP-65 firmware many times does not give the angle to the nearest second in f→D.MS conversion. And, of course, it gives *no more* than seconds. But many people need to work to the nearest second and some to finer than that.

Well, **Bob Edelen** of Aurora, Colorado, has found much shorter routines than those previously published in KEY NOTE. If you use these conversions, study the routines carefully. They should do the job for you. And think about Mr. Edelen's final sentence in his letter to us: "Maybe some user has a better and shorter way."

TO D.MS (or H.MS)*

```
LBL      23
A        11
ENTER    41
f        31
→D.MS    03
STO 1    33 01
f⁻¹      32
→D.MS    03
–        51
.        83
3        03
6        06
x        71
RCL 1    34 01
+        61
RTN      24
```

*Must be in degree mode.

FROM D.MS (or H.MS)*

```
LBL      23
B        12
EEX      43
4        04
x        71
f⁻¹      32
INT      83
g LSTx   35 00
f        31
INT      83
EEX      43
4        04
÷        81
f⁻¹      32
→D.MS    03
g x→y    35 07
3        03
6        06
0        00
0        00
÷        81
+        61
RTN      24
```

*Must be in degree mode.

Machine Design On The HP-65

Here is the abstract of a program (actually, three programs in one) that was featured in KEY NOTE over a year ago. We present it again because the "B" revision makes it *enormously* better than it already was in the original version. If you are remotely interested in this subject, or in machine design in general, you'll want to investigate this program. And, as an added attraction, the documentation is among the very best in the Library.

Synchronous Belt Indexer (Order #03801B)*

These iterative programs compute the geometric parameters for incremental motion transmissions, using a constant angle driver such as a stepping motor or Geneva drive mechanism, synchronous (toothed) belt and pulleys, chain and sprockets, or gear and rack. Examples commonly used include: production-line work-positioners, punched-card/paper-tape indexers, and step-and-repeat mechanisms. Program 1 solves step-angle/pulley-size combinations with linear/step increment within user-defined limit-of-error. Prgrams 2 and 3 include an Error Warning for center distances less than the minimum center distance.

Author: **Thomas Hender**
Corvallis, Oregon

In European areas, order by number 51520A.

More COGO Comments

Several issues ago we printed an article about **Carl M. King's** "COGO-6500 Series of Coordinate Geometry Calculations for Surveyors." Some of you purchased these programs and made comments that the programs were written to accept *azimuth only* and not bearing and distance directly. So we asked Mr. King to write a reply and, so, for the benefit of our U.S. and Canadian readers, here is that letter.

I have endeavored to make the COGO programs as instructive and as easy to use as possible, within the limitations of a very small yet remarkably capable little computer. I call it a "computer" because it is programmable and is able to carry out relatively complicated instructions.

The COGO programs provide you with a virtually complete system of surveying calculations. The North-South, East-West system of coordinates provides the mapmaker with a precise method of plotting points on the flat drawing board. However, the measurements obtained by the surveyors in the field are obtained in a different form. Through the medium of the COGO programs we are able to take the raw data and convert it for the mapmaker. We are able to manipulate the data back and forth in a twinkling, and discover useful relationships, such as points of intersection, distances between intersections, and slopes of lines, all necessary and useful to the mapmaker.

In designing such a system the problem is two-fold. We not only have to program the computer, but we have to "document" the program, which means we have to provide a training course for the User. The more complex a program is, the more extensive must be the documentation. In this particular case there is a communication gap in regards to the reading of angular directions or "headings." A "heading" is a general term for the direction in which you are proceeding. Headings can take several different forms depending on your habits and background. However, you need to communicate with a little handheld calculator, and it is much more economical and more convenient for you, in the long run, to learn to speak its language than it would be to program it to speak yours.

In the case of "bearings" VS "azimuths" we have several things going for us:

1. They are both expressed in the same units.
2. At least half of the time they both use North as the starting point.
3. The calculator interprets plus (+) and minus (−) signs in a significant and predictable fashion.

It is this item 3 that gives the clue and the cue for communicating with the calculator. A bearing actually consists of an angle and a distance. You may attach a (+) or a (−) sign to the angle, and you may attach a (+) or a (−) sign to the distance. This provides four combinations, and enables you to specify the quadrant while entering the *actual numerical bearings* into the calculator.

Using a negative distance may seem unnatural to you at first, but it actually works. Just keep in mind that CCW angles are negative; and, for the S'ly quadrants, oppositely facing distances are negative. (The calculator calculates *all* its angles from the NORTH.) The qualified surveyor will be quick to realize that going SW, for example, is just the opposite of going NE, and giving the calculator a negative distance is all that is required to designate the difference. I pointed this out in the COGO instructions in the form of a HINT. Maybe I should have enlarged on it a little more.

The other alternative of identifying the quadrant by number would have required the utilization of numerous program steps, and would have robbed the COGO programs of much of their convenience.

A Positive Error?

In the last KEY NOTE, on page 4, there was a small error in the article "Are Flags (+) or (−)?" In the second line of the table, in the third column, there's a notation: f^{-1} TF2 (+). It was quite obvious that the (+) should have been a (−), and we feel sure that most people realized that when they read the article. We hope the error did not cause any confusion.

A Better Way To Do Tan^{-1}

This programming idea came from **Tak Y. Lee** of Wellesley, Massachusetts. If you haven't already discovered it by yourself, you'll find it very useful.

Dear Editor:

The HP-65 has no way to distinquish $\tan^{-1}$ ($-$y/x) from $\tan^{-1}$ (y/$-$x) if the arc tangent function is done with the f^{-1} TAN keys. A better way to do it is to use the f R→P keys as follows:

(y including sign)
ENTER
(x including sign)
f
R→P
g R↓

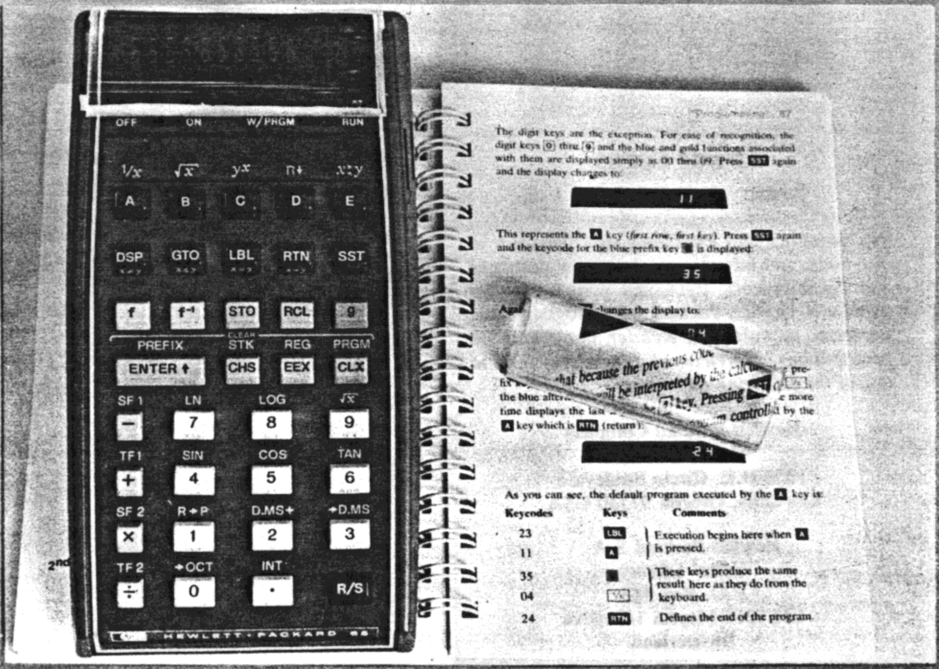
This way, for example, $\tan^{-1}$ ($-$1/1) = −45° and $\tan^{-1}$ (1/$-$1) = 135°, but $\tan^{-1}$ ($-$1) is always −45° when you use the f^{-1} TAN keys.

Bigger Than Life!

If special requirements or applications make it necessary for you to photograph the display on your calculator, you might have some difficulty recording (on black and white film) the red LED numbers. Particularly when you also want to include the keyboard or the entire calculator. When you move the camera back in order to frame the entire calculator, the display is reduced in size, and the display is less brilliant because of the increased distance.

Mr. **W.W. Trotti** of Cayce, South Carolina, has found one solution to the problem. As you can see in the accompanying photograph, his modification greatly magnifies the display and does not require altering the calculator. He used a Bausch & Lomb reading glass (which is made of clear plastic), cut it on a bandsaw to fit the HP-65, and mounted it over the display with double-backed (double-stick) adhesive tape.

Although Mr. Totti's "magnifier" does work and will double the size of the display, it does significantly cut down on the viewing angle at which you can easily read the displayed numbers. So, unless you need the increased size for photographic purposes or because of failing eyesight, you really don't *need* this modification. Also, another caution is that you cannot get the calculator in its case if you leave the magnifier on it.



(continued from page 2)

Bell-Fruit Slot Machine (00218D)*

A much-improved version of the HP-65 ''Bell-Fruit'' program, this program will simulate a Mills brand, 10-cent slot machine, both in standard payoffs as well as in precisely duplicating the same odds in which any particular combination will occur. It includes automatic recall of winnings by one key, a delayed action tumbler display, and a prestored seed for quick game start-up. (112 steps)

Author: **Craig A. Pearce**
Berwyn, Illinois

**In European areas, order by number 00202D.*

Now, here is a series of five ''word-game'' programs from the inimitable word-game mastermind, **John R. Rausch,** of Franklin, Ohio. Many of Mr. Rausch's HP-65 word-game programs were best sellers, and we are sure you will find these equally—probably even *more*–interesting. All of these programs utilize a keyboard overlay for alphabetic instead of numeric entries, and they are exceptionally well-documented and legible. And before you rush to order any *one* program, read the following abstracts carefully; some programs are dependent on others—as subroutines. In fact, you should order the ''Word Game Subroutine'' if you are interested in *any* of the programs.

Word Encoder (#00253D)*

This program is used to encode words into a string of up to 50 positions and to write this string onto a data card to be used by such word games as ''Word Bagles'' (program #00255D), ''Probe'' (#00256D), and ''Hangman'' (#00257D). If you intend to write your own word game using this program to encode the words, you should also order the ''Word Game Subroutine'' (#00254D) to decode them. (139 steps)

**In European areas, order by number 00196D.*

Word Game Subroutine (#00254D)*

The purpose of this subroutine is to interface word game programs with data-card words created by the program titled ''Word Encoder'' (program #00253D). Using this subroutine allows the programmer to concentrate on the mechanics of the game without having to program the entry of encoded words into his logic. One use of this subroutine may be found in the ''Hangman Word Game'' (#00257D). (90 steps)

**In European areas, order by number 00197D.*

Word Bagles (#00255D)*

''Word Bagles'' is a game in which you try to guess a three- to five-letter word. When you enter your guess word, you are given a clue that tells the number of letters you have guessed in their correct position and the number of letters you have guessed but have in the wrong position. This program also uses cards created by the ''Word Encoder'' program (#00253D). *It will not function without the ''Word Game Subroutine''* (#00254D), which is not included and must be purchased separately. (194 steps)

**In European areas, order by number 00198D.*

Probe Word Game (#00256D)*

''Probe'' is a two-player game in which players compete against each other while trying to guess a word one letter at a time. Points are awarded for correct guesses. The word can have blanks on either end; however, a penalty is given for an incorrect blank guess. This program uses data cards created by the ''Word Encoder'' program (#00253D). *It will not function without the ''Word Game Subroutine''* (#00254D), which is not included and must be purchased separately.

**In European areas, order by number 00199D.*

Hangman Word Game (#00257D)*

This program uses words on a data card created by the ''Word Encoder'' program (#00253D) to play ''Hangman'' with you. After being told the number of letters, you try to discover the word by guessing letters. After each guess you are told the positions in the word in which that letter appears, plus the number of wrong guesses made so far. *It will not function without the ''Word Game Subroutine''* (#00254D), which is not included and must be purchased separately.

**In European areas, order by number 00200D.*

HP KEY NOTES
January 1977 Vol. 1 No. 1

Programming and operating tips, answers to questions, and information about new programs and developments. Published periodically for owners of Hewlett-Packard fully programmable personal calculators. *Reader comments or contributions are welcomed. Please send them to one of the following addresses.*

Hewlett-Packard Company
Users' Library
1000 N.E. Circle Boulevard
Corvallis, Oregon 97330 USA

Hewlett-Packard S.A.
Users' Library
P.O. Box 349
CH-1217 Meyrin 1/Geneva
Switzerland

Printed in U.S.A.